

LEARNING A NONLINEAR GENE REGULATION MODEL FROM PERTURBED STEADY-STATE DATA

BY ARWEN MEISTER, YE (HENRY) LI, BOKYUNG CHOI, AND WING HUNG WONG

Stanford University

Abstract Biological structure and function depend on complex regulatory interactions between many genes. A wealth of gene expression data is available from high-throughput genome-wide measurement technologies, but effective gene regulatory network inference methods are still needed. Model-based methods founded on quantitative descriptions of gene regulation are among the most promising, but many such methods still rely on ad hoc inference approaches and lack experimental interpretability. We propose an experimental design and develop an associated statistical method for learning a quantitative, interpretable, predictive, biophysics-based ordinary differential equation model for gene regulation. We fit the model parameters using gene expression measurements from perturbed steady-states of the system, like those following overexpression or knockdown experiments. Although the original model is nonlinear, our design allows us to transform it into a convex optimization problem by restricting attention to steady-states and using the lasso for parameter selection. Here, we describe the model and inference algorithm and apply them to a synthetic six-gene system, demonstrating that the model is detailed and flexible enough to account for activation and repression as well as synergistic and self-regulation, and that the algorithm can efficiently and accurately recover the parameters used to generate the data.

Introduction. Complex interactions between many genes give rise to biological structure and function that sustain life. The Central Dogma (Jacob and Monod, 1961; Crick, 1970) provides a qualitative description of how these processes occur, but precise quantitative modeling is still needed (Tyson, Chen and Novak, 2003; Rosenfeld, 2011). Research into the detailed mechanisms of gene expression over the past few decades has shown that expression is regulated by a complex system of gene interactions. Recently, microarray and sequencing technologies (DeRisi, Iyer and Brown, 1997; Ren et al., 2000; Robertson et al., 2007; Mortazavi et al., 2008) have enabled high-throughput genome-wide expression level measurements. This data enables detailed study of gene networks (Holstege et al., 1998; Lee et al., 2002; Tegner et al., 2003; Segal et al., 2003; Bar-Joseph et al., 2003; Hu, Killion and Iyer, 2007; Zhou et al., 2007). The goal is to understand how genes interact to give rise to the biochemical complexity that allows organisms to live, grow and reproduce.

Gene expression measurements contain information useful for reconstructing the underlying interaction structure (DeRisi, Iyer and Brown, 1997; Holstege et al., 1998; Hughes et al., 2000) because gene regulatory systems have a defined ordering (Avery and Wasserman, 1992), forming pathways that ultimately connect to form networks (Alon, 2007; De Smet and Marchal, 2010). At the turn of the century, researchers began applying statistical tools to genome-wide expression data to understand complex gene interactions. Eisen et al showed that genes from the same pathways and with similar functions cluster together by expression pattern (Eisen et al., 1998). Soon after, module-based network inference methods appeared, in which co-expressed genes are grouped into the same

module of cellular function (Segal et al., 2003; Bar-Joseph et al., 2003). More recently, methods based on descriptive but non-mechanistic mathematical models (Gardner et al., 2003; Tegner et al., 2003; Bansal et al., 2007; Faith et al., 2007; Friedman, 2004) have gained prominence. Model-based methods provide a quantitative description of gene regulation and can be used to predict the future behavior of the system. However, most of these methods still rely on ad hoc inference approaches that depend on various assumptions to find interactions between genes without learning experimentally interpretable parameters. In this paper, we extend the current model-based literature by proposing a statistical method for inferring a mathematical model for gene regulation.

Our method is based on a quantitative, experimentally interpretable biophysics-based ordinary differential equation (ODE) model of gene regulation. We focus on transcriptional regulation since transcription lies at the core of gene expression regulation (Holstege et al., 1998; Hu, Killion and Iyer, 2007). A standard biophysics approach to gene transcription uses a thermodynamic model of transcription factor (TF) binding to the gene promoter. TF binding can either activate or repress transcription. Many models of this type have been proposed, but we believe that the Bintu et al model is standard. The idea traces back to the beginning of systems biology in the biophysics field (Ackers, Johnson and Shea, 1982; Shea and Ackers, 1985; von Hippel et al., 1974), although it has been used only for modeling and not for reconstruction.

The dynamical model described in Bintu et al is a rich, flexible and interpretable physics-based model. Their ODE-based model of gene expression is based on the thermodynamics of transcription factor and RNA polymerase binding to gene promoters. The form of the model is flexible enough to capture the full range of gene regulatory behavior in quantitative detail, and the parameters are physically interpretable. Another notable thermodynamic model is that of the annual DREAM competition, but it has many biochemical assumptions and model parameters, like the Hill coefficient of transcription factor binding events, that cannot be estimated using gene expression measurements, so the network reconstruction requires ad hoc inference methods to learn the underlying gene interactions (Yip et al., 2010; Pinna, Soranzo and de la Fuente, 2010; Marbach et al., 2010; Schaffter, Marbach and Floreano, 2011). Thus, we choose to base our approach on the Bintu model since it is simple, captures the mechanism of transcription well, and lends itself to statistical network inference.

We have developed an approach to learn gene network structure by inducing perturbed steady-states and fitting the parameters in the thermodynamic model of Bintu et al using a set of gene expression measurements. Although the original inference problem is nonlinear, we can transform it into a convex optimization problem by restricting our attention to steady-states. We use the lasso (Tibshirani, 1996) for parameter selection. As a proof of principle, we test the method on a simulated embryonic stem cell (ESC) transcription network (Chickarmane and Peterson, 2008) given by a system of ODEs based on the Bintu et al model. Here, we demonstrate that the inference algorithm is computationally efficient, accounts for synergistic regulation and self-regulation, and recovers the correct parameters used to generate the data. Furthermore, the method only requires a set of steady-state gene expression measurements. Experimental researchers in the biological sciences can use this method to infer gene networks in a much more principled, detailed manner than earlier approaches allowed.

Dynamical Systems Model. We model gene expression regulation as a dynamical system. Let $x \in \mathbb{R}^n$ represent RNA concentrations and $y \in \mathbb{R}^n$ represent protein concentrations corresponding

to a set of n genes. We assume that the production rate of the RNA transcript x_i of gene i is proportional to the probability that RNA polymerase (RNAP) is bound to the promoter, the production rate of protein product y_i of gene i is proportional to the concentration of the RNA transcript x_i , and that both the RNA transcript and protein products of gene i degrade at fixed rates $(\lambda_i^{RNA}, \lambda_i^{Protein})$. The probability that RNA polymerase binds to the promoter is modeled as a nonlinear function f of y , since RNAP binding is regulated by a set of transcription factors.

$$(1) \quad \begin{aligned} \frac{dx_i}{dt} &= \tau_i f_i(y) - \lambda_i^{RNA} x_i \\ \frac{dy_i}{dt} &= r_i x_i - \lambda_i^{Protein} y_i \end{aligned}$$

Based on the thermodynamics of RNAP and TF binding, we assume the following form for f_i (Bintu et al., 2005b,a):

$$(2) \quad f_i(y) = \frac{b_{i0} + \sum_{j=1}^m b_{ij} \prod_{k \in S_{ij}} x_k}{1 + \sum_{j=1}^m c_{ij} \prod_{k \in S_{ij}} y_k}$$

where S_{ij} lists the gene products that interact to form a regulatory complex, and b_{ij}, c_{ij} are non-negative coefficients that must satisfy $c_{ij} \geq b_{ij} \geq 0$. The coefficients b_{ij} and c_{ij} depend on the binding energies of regulator complexes to the promoter. b_{i0} and c_{i0} correspond to the no-regulator case ($\prod_{k \in S_{i0}} y_k = 1$), and the coefficients are normalized so that $c_{i0} = 1$. Details and a derivation are given in Appendix A.

The form of f_i allows us to model the full spectrum of regulatory behavior in quantitative detail. Terms that appear in the denominator only are repressors, and the degree of repression depends on the magnitude of the coefficient, while terms that appear in the numerator and denominator may act as either activators or repressors depending on the relative magnitudes of the coefficients and the current gene expression levels. Terms may represent either single genes or gene complexes. The model can even be extended to account for environmental factors that affect gene regulation, though we will not discuss it further here.

As an example, consider the simple two-gene network shown in Figure 1. Suppose that genes 1 and 2 have RNA concentrations x_1, x_2 , and protein concentrations y_1, y_2 , respectively, and that gene 1 is activated by protein 2 and repressed by its own product (protein 1), while gene 2 is repressed by a complex formed by proteins 1 and 2. The situation corresponds to the following equations:

$$(3) \quad \begin{aligned} \frac{dx_1}{dt} &= \tau \frac{b_{10} + b_{11}x_2}{1 + c_{11}x_2 + c_{12}x_1} - \lambda_1^{RNA} x_1, & \frac{dy_1}{dt} &= r_1 x_1 - \lambda_1^{Protein} y_1 \\ \frac{dx_2}{dt} &= \tau \frac{b_{20}}{1 + c_{21}x_1 x_2} - \lambda_2^{RNA} x_2, & \frac{dy_2}{dt} &= r_2 x_2 - \lambda_2^{Protein} y_2 \end{aligned}$$

In the notation above, we have $S_{11} = \{2\}, S_{12} = \{1\}, S_{21} = \{1, 2\}$. The parameters $b_{10}, b_{11}, c_{11}, \dots$ determine the magnitude of the repression or activation. As this example shows, the model is flexible enough to capture a wide range of effects, including self-regulation (that is, regulation of a gene by its own protein product, most commonly as repression) and synergistic regulation by protein complexes (two or more proteins bound together to form a regulatory unit), in quantitative detail. Furthermore, the model is predictive: if we know or can infer the coefficients in the model, we can predict the future behavior of the system starting from any initial condition.

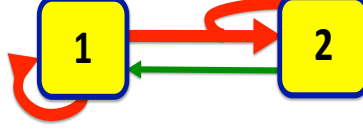


Figure 1: Simple two-gene network example described by equation 3 (with parameters $b_{11} = c_{11} = 0.1$ is typical for activators, $c_{12} = 10$ for repressors, and $b_{10} = 0.01$ for constants in the numerator). Gene 1 is activated by the protein product of gene 2 and repressed by its own product (an example of self-regulation). Gene 2 is repressed by a complex formed by the product of gene 1 and its own product (synergistic self-regulation). In the diagram, the edge colors indicate activation (green) or repression (red) and the edge weights indicate coefficient sizes, with typical sizes shown in the figure.

Inference problem. The model given by equations 1 and 2 fully describes the evolution of RNA and protein levels and provides a comprehensive, quantitative model of gene regulation, provided we know the parameters. Unfortunately, b_{ij}, c_{ij} and extremely difficult to measure, as they depend on binding energies of RNAP and TFs to the gene promoter. The sheer number of measurements required to characterize all possible TFs (both individual proteins and complexes) also makes this approach infeasible. Therefore, our goal is to use a systems level approach to fit the model using RNA expression data. Specifically, we will assume that $\tau_i, \lambda_i^{RNA}, \lambda_i^{Protein}$ are known or can be measured (alternatively, we can simply absorb these terms into the coefficients b_{ij}, c_{ij} and ignore them). Our data will be measurements of the RNA concentrations x at many different cellular steady states (which correspond to steady-states of the dynamical system). The problem is to infer the values of the coefficients b_{ij}, c_{ij} .

Linear constraints at steady-state. The key to solving this problem efficiently is to restrict our attention to steady-states, as proposed by Choi (Choi, 2012). This allows us to transform a nonlinear ODE fitting problem into a linear algebra problem. A steady-state of the system is one in which RNA and protein levels are constant: $\frac{dx_i}{dt} = \frac{dy_i}{dt} = 0$. Steady-states of the system correspond to cell states with roughly constant gene expression levels, like embryonic stem cell, skin cell or liver cell. In contrast, an embryonic stem cell in the process of differentiating is not in steady state. Perturbed steady-states are particularly interesting. After a perturbation like gene knockdown, a cell's gene expression levels are in flux for some time while they adjust to the change. Eventually, if it is still viable, the cell may settle to a new steady-state. These perturbed steady-state are especially helpful for understanding gene regulation.

In our model, the steady-state conditions $\frac{dx_i}{dt} = \frac{dy_i}{dt} = 0$ mean that:

$$0 = \tau_i f_i(y) - \lambda_i^{RNA} x_i, \quad 0 = r_i x_i - \lambda_i^{Protein} y_i \implies y_i = \frac{r_i x_i}{\lambda_i^{Protein}}$$

Defining $\tilde{f}_i(z) = f_i(\frac{r_i}{\lambda_i^{Protein}} z)$ yields

$$0 = \tau_i \tilde{f}_i(x) - \lambda_i^{RNA} x_i,$$

Absorbing the constants into the coefficients b_{ij}, c_{ij} , (so that $\tilde{b}_{ij} = b_{ij} \prod_{k \in S_{ij}} \frac{r_k}{\lambda_k^{Protein}}, \tilde{c}_{ij} = c_{ij} \prod_{k \in S_{ij}} \frac{r_k}{\lambda_k^{Protein}}$) we obtain the final equation

$$\tau_i \frac{b_{i0} + \sum_j b_{ij} \prod_{k \in S_{ij}} x_k}{1 + \sum_j c_{ij} \prod_{k \in S_{ij}} x_k} - \gamma_i x_i = 0,$$

4

or

$$\tau_i(b_{i0} + \sum_j b_{ij}\Pi_{k \in S_{ij}}x_k) - \gamma_i x_i(1 + \sum_j c_{ij}\Pi_{k \in S_{ij}}x_k) = 0,$$

(by multiplying both sides by the denominator). The last equation is linear in the coefficients b_{ij}, c_{ij} ! In order to solve for b_{ij}, c_{ij} , we will need to collect many different expression measurements x at both naturally occurring and perturbed steady-states. Each steady-state measurement will lead to a different linear equation. These can be arranged into a linear system that we can solve for the coefficients.

Problem formulation. Our problem is to find b_{ij}, c_{ij} such that

$$0 = \tau_i(b_{i0} + \sum_j b_{ij}\Pi_{k \in S_{ij}}x_k^{(m)}) - \gamma_i x_i(1 + \sum_j c_{ij}\Pi_{k \in S_{ij}}x_k^{(m)}), \quad \forall m = 1, \dots, M,$$

given RNA expression data $x^{(m)}$ at many different steady-state points $m = 1, \dots, M$ and known translation and degradation rates $\tau_i, \lambda_i^{RNA}, \lambda_i^{Protein}$. We solve a separate problem for each gene i , since the coefficients b_{ij}, c_{ij} in the differential equation $dx_i/dt = \dots$ for gene i are independent of the coefficients in the differential equations for other genes. Since we cannot know ahead of time which potential regulatory terms $\Pi_{k \in S_{ij}}x_k$ are actually involved, we include all possible terms up second-order and look for sparse b_{ij}, c_{ij} , interpreting $c_{ij} = 0$ to mean that term $\Pi_{k \in S_{ij}}x_k$ is not a regulator of gene i .

Consider gene 2 in the two-gene example. Suppose we have expression measurements for a naturally occurring steady state (x_1^0, x_2^0) , and a perturbed steady-state following gene 1-knockout $(0, x_2^1)$. We obtain two linear equations in the coefficients b_{20}, c_{21} :

$$\begin{aligned} \tau b_{20} - \lambda_2 x_2^0(1 + c_{21}x_1^0 x_2^0) &= 0, & (\text{steady-state } (x_1^0, x_2^0)) \\ \tau b_{20} - \lambda_2 x_2^1 &= 0, & (\text{steady-state } (0, x_2^1)) \end{aligned}$$

If we knew a priori that complex $x_1 x_2$ was the only regulator of gene 2, these two equations would allow us to solve for the coefficients ($b_{20} = \frac{\lambda_2 x_2^1}{\tau}$, $c_{21} = \frac{x_2^0 - x_2^1}{(x_2^0)^2}$). Typically we do not know the regulators beforehand, however, and we need to use the data to identify them. That is, we include all possible terms (up to second order) in the equation:

$$\tau(b_{20} + b_{21}x_1x_2 + b_{22}x_1 + b_{23}x_2) - \lambda_2 x_2(1 + c_{21}x_1x_2 + c_{22}x_1 + c_{23}x_2) = 0.$$

and estimate sparse coefficients b_{ij}, c_{ij} using several steady-state measurements $(x_1^{(m)}, x_2^{(m)})$. (We should find that the recovered coefficients $b_{21}, b_{22}, b_{23}, c_{22}, c_{23}$ are very close to zero, since the corresponding terms do not appear in the true equation.)

We can compactly express the general system above by defining z_i as the vector with entries $z_i(j) = \Pi_{k \in S_{ij}}x_k$ (with the convention that $z_i(0) = 1$, $z_i(j) = x_j$ for $j = 1, \dots, n$), which yields

$$0 = \tau_i b_i^T z_i - \gamma_i z_i(i) c_i^T z_i, \quad \forall m = 1, \dots, M.$$

If we form a matrix G_i by concatenating the row vectors $z_i^{(1)}, \dots, z_i^{(M)}$ and let D_i be a diagonal matrix with entries $z_i^{(m)}(i)$, $m = 1, \dots, M$, we can express this as

$$\begin{bmatrix} G_i & -\gamma D_i G_i \end{bmatrix} \begin{bmatrix} b_i \\ c_i \end{bmatrix} = 0.$$

with the constraints $0 \leq b_i \leq c_i$, $c_i(0) = 1$

Algorithm. We need to solve the linear system

$$\begin{bmatrix} G_i & -\gamma D_i G_i \end{bmatrix} \begin{bmatrix} b_i \\ c_i \end{bmatrix} = 0.$$

for b_i, c_i , subject to the constraints $0 \leq b_i \leq c_i$, $c_i(0) = 1$. In order to account for measurement noise and encourage sparsity in b_i, c_i (since we know that each gene has only a few regulators), we will minimize the ℓ_2 -norm error with ℓ_1 regularization (Tibshirani, 1996). This leads to the convex optimization problem

$$\begin{aligned} & \text{minimize} \quad \left\| \begin{bmatrix} G_i & -\gamma D_i G_i \end{bmatrix} \begin{bmatrix} b_i \\ c_i \end{bmatrix} \right\|_2 + \lambda(\|b_i\|_1 + \|c_i\|_1) \\ (4) \quad & \text{subject to} \quad 0 \leq b_i \leq c_i, \quad c_i(0) = 1, \end{aligned}$$

where λ is a parameter controlling sparsity that we can choose using cross validation. Since the problem is convex, it can be solved very efficiently even for large values of n and m .

Nonidentifiability. Our model’s ability to capture self-regulation is very powerful, but it also leads to a particular form of nonidentifiability. For certain forms of the equation, given only steady-state measurements, it can be impossible to determine whether self-regulation is either completely absent or present in every term. Specifically, any valid equation of the form:

$$(5) \quad \frac{dx_i}{dt} = \frac{b_{i0} + \sum_{j=1}^N b_{ij} \Pi_{k \in S_{ij}} x_k}{1 + \sum_{j=1}^N c_{ij} \Pi_{k \in S_{ij}} x_k} - \gamma_i x_i, \quad b_{i0} < 1$$

is indistinguishable at steady-state from any member of the following family of valid equations indexed by the constant c :

$$(6) \quad \frac{dx_i}{dt} = \frac{(cb_{i0} + \gamma_i)x_i + \sum_{j=1}^N cb_{ij} \Pi_{k \in S_{ij}} x_i x_k}{1 + cx_i + \sum_{j=1}^N cc_{ij} \Pi_{k \in S_{ij}} x_i x_k} - \gamma_i x_i, \quad c \geq \frac{\gamma}{1 - b_{i0}}$$

We will refer to these as the ‘simple’ and ‘higher-order’ forms of the equation, respectively. The short proof of their equivalence is given in Appendix B. The condition $c \geq \frac{\gamma}{1 - b_{i0}}$ guarantees that $c > 0$ and $0 \leq cb_{i0} + \gamma_i \leq c$ (since $0 \leq b_{i0} < 1$) and $0 \leq cb_{ij} \leq cc_{ij}$ (since $0 \leq b_{ij} \leq c_{ij}$).

We can distinguish between these two alternative forms by measuring the derivative of the concentration away from steady-state and comparing it to the derivative predicted by each form of the equation. This requires only a few extra thoughtfully-selected measurements. The details are in Appendix C.

Simulated six-gene subnetwork in mouse ESC. To demonstrate the inference approach, we apply our method to a synthetic six-gene system based on the Oct4, Sox2, Nanog, Cdx2, Gcnf, Gata6 subnetwork in mouse embryonic stem cell (ESC). Chickarmane et al developed this system based on a synthesis of knowledge about ESC gene regulation accumulated over the past two decades (Chickarmane and Peterson, 2008). The network structure is shown in Figure 3a, and the detailed model is given by the following system of ODEs in the six genes:

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{0.001 + [A] + 0.005[O][S] + 0.025[O][S][N]}{1 + [A] + 0.001[O] + 0.005[O][S] + 0.025[O][S][N] + 10[O][C] + 10[Gc]} - 0.1[O] \\
\frac{d[S]}{dt} &= \frac{0.001 + 0.005[O][S] + 0.025[O][S][N]}{1 + 0.001[O] + 0.005[O][S] + 0.025[O][S][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.001 + 0.1[O][S] + 0.1[O][S][N]}{1 + 0.001[O] + 0.1[O][S] + 0.1[O][S][N] + 10[O][G]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{0.001 + 2[C]}{1 + 2[C] + 5[O][C]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.001 + 0.1[C] + 0.1[G]}{1 + 0.1[C] + 0.1[G]} - 0.1[Gc] \\
(7) \quad \frac{d[G]}{dt} &= \frac{0.1 + [O] + 0.00025[G]}{1 + [O] + 0.00025[G] + 15[N]} - 0.1[G]
\end{aligned}$$

This model has many of the same qualitative characteristics as the biological mouse ESC network ([Chickarmane and Peterson, 2008](#)). In particular, the system can support the four different steady-states: embryonic stem cell (ESC), differentiated stem cell (DSC), endoderm and trophectoderm, and can switch between them when certain genes' expression levels are changed. In the Oct4 equation, A represents an external activating factor, whose concentration $[A]$ depends on the culture condition. Each of the four steady-states has a corresponding value of $[A]$: 10 for ESC and DSC, 25 for endoderm, and 1 for trophectoderm. For the remainder of this paper, we will regard $[A]$ as known. The explicit system ODEs (7) allows us to generate data to fit our model and also to quantitatively compare our recovered solution to the ground truth. The qualitative similarity of this synthetic network to a real biological network gives us confidence that the results of this experiment are likely to translate well to real biological networks.

We observe that the $Cdx2$, $Gcnf$ and $Gata6$ equations have alternative forms (provided we ignore the very small constant term in the $\frac{d[C]}{dt}$ equation and $[G]$ term in the $\frac{d[G]}{dt}$). With the minimum possible value of c , the alternative forms are:

$$\begin{aligned}
\frac{d[C]}{dt} &= \frac{0.95}{1 + 2.5[O]} \quad (c = 2) \\
\frac{d[Gc]}{dt} &= \frac{0.1001[Gc] + 0.01[C][Gc] + 0.01[Gc][G]}{1 + 0.1[Gc] + 0.01[C][Gc] + 0.01[Gc][G]} - 0.1[Gc] \quad (c = 0.1) \\
(8) \quad \frac{d[G]}{dt} &= \frac{0.111[G] + 0.111[O][G]}{1 + 0.111[G] + 0.111[O][G] + 1.67[N][G]} - 0.1[G] \quad (c = 0.111)
\end{aligned}$$

To resolve this, we will apply our method twice, once allowing self-regulation and again disallowing it. Then we will compare the two recovered forms of each equation and quality of the fits to determine whether nonidentifiability exists in each case. If so, we will break the tie by examining derivatives.

To fit the model, we collect data on the the expression levels of all six genes at many different steady-states. First we measure the expression levels at all four wildtype steady states: SC, DSC, endoderm and trophectoderm. We also induce additional perturbed steady-states by simulating knockdowns and knockups of each gene. For a knockdown, we hold a gene at one fifth of its steady-state expression level; for a knockup we hold a gene at a twice its steady-state level. In each case

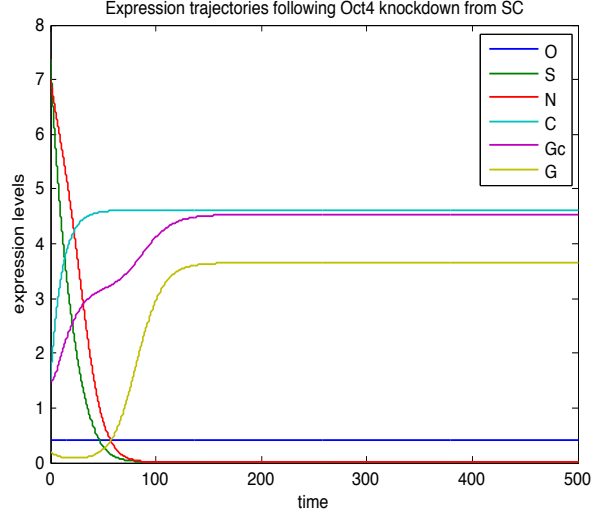


Figure 2: Gene expression trajectories during an Oct4 knockdown from SC steady-state. The expression of Oct4 is artificially reduced to 20% of its SC steady-state expression level and held there, causing the expression levels of the targets of Oct4 to change in response, which in turn impact their targets. The system eventually reaches a new steady-state different from SC. We measure the vector of expression levels at the new steady-state and use it as data in the inference algorithm. Since Oct4 is knocked down, this induced steady state does not provide useful information about the Oct4 equation, but it is useful for understanding the role of Oct4 and other genes in the equations of the remaining five genes.

we wait for the system to settle to a new steady-state, then measure the expression levels. Figure 2 shows the expression trajectories during Oct4 knockdown from the ESC steady-state as an example.

The details of the simulation are given in Appendix D. We begin by testing the algorithm on noiseless data. We solve the optimization problem 4 once, then we solve it again with additional constraints prohibiting self-regulation. In each case, we use cross validation to select the sparsity parameter λ (Figure 5). The quality of the fit is comparable for the latter three equations whether we restrict self-regulation or not, while for the first three equations restricting self-regulation has a significant negative impact on the fit (Table 1), indicating that the first three equations are unambiguous while the last three have two possible forms. To resolve the nonidentifiability in the latter three equations, we measure the derivatives of Cdx3, Gcnf and Gata6 immediately after some additional informative perturbations: Oct4, Cdx2 and Nanog knockouts, respectively (Figure 6). The test reveals that that Gcnf and Gata6 have the simple form, while Cdx2 has a higher-order

form. This procedure allows for near-perfect recovery:

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{0.001 + [A] + (0.005[O][S] + 0.025[O][S][N])}{1 + [A] + (0.001[O] + 0.005[O][S] + 0.025[O][S][N]) + 10[O][C] + 10[Gc]} - 0.1[O] \\
\frac{d[S]}{dt} &= \frac{0.001 + 0.005[O][S] + 0.025[O][S][N]}{1 + 0.005[O][S] + 0.025[O][S][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.1[O][S] + 0.1[O][S][N]}{1 + 0.1[O][S] + 0.1[O][S][N] + 10[O][G]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{2[C]}{1 + 2[C] + 5[O][C]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.001 + 0.1[C] + 0.1[G]}{1 + 0.1[C] + 0.1[G]} - 0.1[Gc] \\
(9) \quad \frac{d[G]}{dt} &= \frac{0.1 + [O]}{1 + [O] + 0.03[N][Gc] + 15[N]} - 0.1[G]
\end{aligned}$$

Next we add zero-mean Gaussian noise to each measurement, with standard deviation 1% of the measurement magnitude. We use the same steady-states as in the noiseless case, plus knockup-knockdown of each pair of genes starting from ESC and DSC. Using a similar approach (detailed in Appendix D), we recover:

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{[A]}{1 + [A] + 9.9[Gc] + 9.9[O][C]} - 0.1[O] \\
\frac{d[S]}{dt} &= \frac{0.001[O][S] + 0.0005[S][N] + 0.025[O][S][N]}{1 + 0.001[O][S] + 0.0005[S][N] + 0.025[O][S][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.09[O][S][N]}{1 + 0.1[G][Gc] + 0.09[O][S][N] + 9.1[O][G]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{2[C]}{1 + 2[C] + 5[O][C]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.1[C] + 0.1[G]}{1 + 0.1[C] + 0.1[G]} - 0.1[Gc] \\
(10) \quad \frac{d[G]}{dt} &= \frac{0.1 + 0.9[O]}{1 + 0.9[O] + 14.2[N]} - 0.1[G]
\end{aligned}$$

In order to produce clean equations and network diagrams, we choose appropriate thresholds for each equation below which we zero the coefficients. We set the thresholds at 0.1% (noiseless case) or 1% (noisy case) of the largest coefficient recovered for each equation. For example, the largest recovered coefficient in the $d[G]/dt$ equation is roughly 15 in either case, so we zero the coefficients that fall below 0.015 (noiseless case) or 0.15 (noisy case). The recovered systems of equations shown above reflect these choices. In the noiseless case, relaxing the threshold on the Oct4 equation to 0.01% leads to the recovery of more correct terms, listed in parentheses. For completeness, we also provide receiver operating characteristic (ROC) curves in Figure 4 to show the tradeoff between true positives and false positives at other thresholds. The network diagrams in Figure 3 (b,c) include an edge if the corresponding coefficient is above the threshold, with weights reflecting the size of the coefficients. These diagrams show that the recovery is nearly perfect in the noiseless case: using the gentler threshold for the Oct4 equation we recover all the true edges except for three very weak ones, and return just one small false positive repressor in the Gata6 equation. In the noisy case,

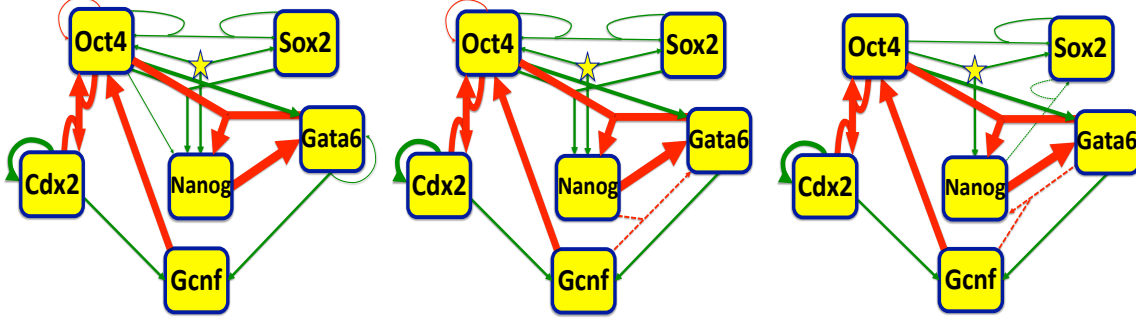


Figure 3: Recovery of a synthetic gene regulatory network based on the biological ESC network using our inference algorithm. The diagrams represent systems of ODEs that quantitatively model the gene interactions. Edge color indicates activation (green) or repression (red), and edge weights correspond to coefficient magnitudes. The arrows point from regulator to target, and self-loops indicate self-regulation. The yellow star represents the third-order complex OSN. (In addition to all possible first- and second-order terms, we allow this special third-order term with a free coefficient.) The left figure represents the original system of ODEs used to generate the data. The center figure shows the network recovered using our inference algorithm on noiseless data, and the right figure shows the recovery with 1% noise added. Both recovered networks reflect coefficient thresholding at 0.1% (noiseless case) or 1% (noisy case) of the largest recovered coefficient in each gene equation (with the exception of the noiseless-case Oct4 equation, thresholded at 0.01% to show the successful recovery of weak edges). The algorithm performs almost perfectly in the noiseless case, except for a false positive repressor on Gata6 and three very weak activation edges missing. In the noisy case, the algorithm recovers all of the strong edges, but misses some of the weaker ones and returns a few small false positives at our chosen thresholding level. Overall, the method captures the major network structure even in the noisy case.

we recover all the large coefficients correctly, although there are a few small false positives and the we miss several of the weakest edges. Overall, the method successfully captures the major network structure even in the presence of low-level noise.

Discussion. Our experiment on the synthetic ESC system demonstrates that our algorithm can accurately recover the coefficients in a complex dynamical systems model of gene regulation and that the method can tolerate low levels of noise. Term selection from among all possible single gene and gene-complex regulators (up to second-degree interactions, plus the third-degree interaction OSN) was successful. The inferred equations are easy to interpret in terms of gene networks, and the detailed quantitative information allows for prediction of future expression trajectories from any starting point.

The approach is also scalable. Since we have formulated our problem as a convex optimization problem (Equation 4), it can be solved efficiently even for large systems using prepackaged software. Furthermore, it is trivially parallelizable, since we need to solve a version of Equation 4 to infer the differential equation coefficients b_{ij}, c_{ij} for each gene i . Parallelization is even more helpful for the cross validation step, where we need to solve Equation 4 for each gene and a sequence of choices sparsity parameter λ . We tested the scalability by running the algorithm with the parallelization discussed above on a simulated 100-gene system. The algorithm ran correctly in a reasonable time frame on a computing cluster.

The high resolution of our model is one of its most valuable features, but it means that accu-

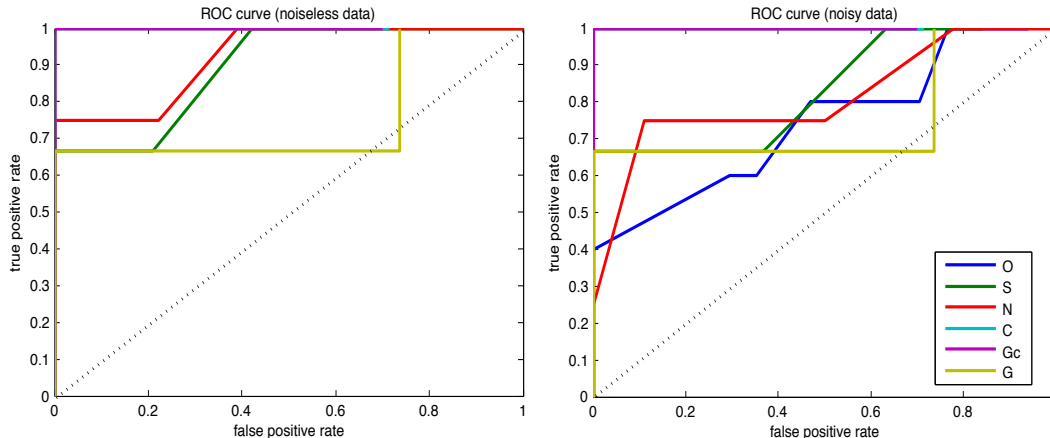


Figure 4: ROC curves for recovered networks from noiseless (right) and noisy (left) data showing the tradeoff between true positive rate (TPR) and false positive rate (FPR) for edge recovery. The ROC curves show the TPR and FPR that result from a range of coefficient threshold choices above which we consider an edge to have been recovered. For the equation $dx_i/dt = \dots$ and threshold t , TPR is defined as the proportion of true edges j with $c_{ij}^{\text{recovered}} > t$ and FPR as the proportion of false edges with $c_{ij}^{\text{recovered}} > t$. For equations with two possible forms, we compare the simple forms of the true and recovered equations. Each gene equation has a different ROC curve as indicated by the legend. The dotted black line is the expected ROC curve for ‘random guessing’ algorithm, while the $(0, 1)$ point corresponds to a perfect algorithm (in fact, our algorithm performs perfectly for the Gcnf equation).

rate term selection may require lots of data, especially in the presence of noise. In our experiment, when we added 1% Gaussian noise, we needed extra data (knockdown/knockup pairs) in order to accurately select terms. When we tried 5% noise, the algorithm consistently selected the large terms in five of the six equations, but we had to add even more data in order to correctly identify the major repressor in the Nanog equation. The Nanog equation is subtle in that Oct4 acts as both an activator in complexes with Sox2 and Nanog and a repressor in a complex with Gata6, so the algorithm tends to select different Gata6 complexes (or the Gata6 singleton) as the major repressor when the data is insufficient. In the 5% noise case, we needed additional data on the role of Gata6 (double-knockdowns and double-knockups of pairs including Gata6 from ESC and DSC) in order to select Oct4-Gata6 as the major repressor of Nanog fairly consistently. Another difficulty we have already discussed is the nonidentifiability that arises from accounting for self-regulation while restricting data to steady-states. Distinguishing between the two possible forms of nonidentifiable equations requires extra derivative data (which is harder to collect experimentally) and extra steps in the algorithm. The constraints on the convex optimization problem (4), which arise from thermodynamic considerations, are sufficient to prevent further nonidentifiability, but in certain cases, certain problems can suffer from near-nonidentifiability of other forms, which may contribute to the challenge of term-selection with noisy or limited data. We ensure accurate term selection by making sure we include enough diverse, high-quality steady-state measurements.

We should also note that our model does not account for the intrinsic noise in gene transcription and translation, although these processes are inherently stochastic, since TF and RNAP binding result from chance collisions between molecules in the cell. However, the stochastic version of our rational-form transcription model is highly complex and there is currently no satisfactory method

for its inference. Studying the deterministic evolution plus additive noise is standard practice for all but linear models of gene expression, and treatment of the deterministic model provides insight into the stochastic model. Here we focus on the additive noise case and leave the study of intrinsic noise for future investigation.

Conclusions. Our model is based on the detailed thermodynamics of gene transcription, and quantitatively captures the full spectrum of regulatory phenomena in a detailed, physically interpretable, predictive manner. Since we can formulate the model fitting problem as a convex optimization problem, we can solve it efficiently and scaleably using prepackaged software. ℓ_1 -regularization allows for term-selection while maintaining the problem convexity. The experiments required to collect the necessary steady-state gene expression data are straightforward to perform, as technologies for knockdowns and knockups are well-established and measuring gene expression is relatively simple. The model accounts for activation and repression by single-protein TFs and synergistic complexes as well as self-regulation, and describes the magnitude of each type of regulation in quantitative detail. Furthermore, the model can be extended to account for environmental effects and auxiliary proteins involved in regulation, including enhancers and chromatin remodelers. The fitted model can accurately predict the evolution of the system from any starting point. Given a set of steady-states gene expression measurements, our algorithm can be used to fit a model which not only predicts further steady-states of the system, but also fully describes the transitions between them. The resulting quantitative model provides unprecedented insight into and control over the entire regulatory lifecycle of the cell.

Appendices

APPENDIX A: THERMODYNAMIC MODEL

In Equation 1, the function $f_i(y)$ represents the probability that RNAP binds to the i th gene promoter. We claim that $f_i(y)$ has the form:

$$f_i(y) \equiv p_{\text{bound}}^{(i)}(y) = \frac{\sum_j e^{-\beta \Delta \epsilon_{ij}^{\text{RNAP}}} P e^{-\beta \Delta \epsilon_{ij}} \prod_{k \in S_{ij}} y_k}{\sum_j (1 + e^{-\beta \Delta \epsilon_{ij}^{\text{RNAP}}} P) e^{-\beta \Delta \epsilon_{ij}} \prod_{k \in S_{ij}} y_k}$$

where $\Delta \epsilon_{ij}$ is the binding energy of the j th complex to the promoter, $\Delta \epsilon_{ij}^{\text{RNAP}}$ is the binding energy of RNAP to the j th promoter-bound complex, and P, x_j are the concentrations of RNAP and gene product j (Bintu et al., 2005a,b).

Any type of regulator (including no regulator at all) can be represented in this framework. For no regulator, we take $S_{ij} = \emptyset$ with the convention that $\prod_{k \in \emptyset} y_k = 1$, set $\Delta \epsilon_{ij} = 0$, and take $\Delta \epsilon_{ij}^{\text{RNAP}}$ as the base binding energy of RNAP to the promoter. For a repressor, $\Delta \epsilon_{ij} < 0$ and $\Delta \epsilon_{ij}^{\text{RNAP}} > 0$; for an activator, $\Delta \epsilon_{ij} < 0$ and $\Delta \epsilon_{ij}^{\text{RNAP}} < 0$.

Setting

$$b_{ij} = e^{-\beta \Delta \epsilon_{ij}^{\text{RNAP}}} P e^{-\beta \Delta \epsilon_{ij}}$$

$$c_{ij} = (1 + e^{-\beta \Delta \epsilon_{ij}^{\text{RNAP}}} P) e^{-\beta \Delta \epsilon_{ij}},$$

we obtain the form given in Section 1:

$$f_i(y) = \frac{b_{ij} \prod_{k \in S_{ij}} y_k}{\sum_j c_{ij} \prod_{k \in S_{ij}} y_k}.$$

Constant terms in the numerator and denominator correspond to the no-regulator case. Letting c_{i0} denote the constant appearing in the denominator, our convention will be to divide all of the coefficients in the numerator and denominator by c_{i0} so that the constant 1 appears in the denominator.

Simplified derivation. The derivation we present here follows Bintu et al and Garcia et al (Bintu et al., 2005a,b; Garcia et al., 2011). For simplicity, we will prove the following claim for the simplified case with one regulator y_1 (as well as the possibility of RNAP binding with no regulator):

$$p_{\text{bound}}^{(i)} = \frac{e^{-\beta \Delta \epsilon_{i0}^{\text{RNAP}}} P + e^{-\beta \Delta \epsilon_{i1}^{\text{RNAP}}} P e^{-\beta \Delta \epsilon_{i1}} y_1}{(1 + e^{-\beta \Delta \epsilon_{i0}^{\text{RNAP}}} P) + (1 + e^{-\beta \Delta \epsilon_{ij}^{\text{RNAP}}} P) e^{-\beta \Delta \epsilon_{i1}} y_1}$$

We will use the following notation: $\epsilon_{P,i1}^S$ is the energy of the state in which RNAP is specifically bound to the regulator-promoter complex, $\epsilon_{P,i0}^S$ is the energy of the state in which RNAP is specifically bound to the promoter without the regulator, ϵ_P^{NS} is the energy when RNAP is bound to a nonspecific binding site, ϵ_{i1}^S is the energy when y_1 is specifically bound to the promoter, and ϵ_{i1}^{NS} is the energy when y_1 is bound to a nonspecific binding site. Then

$$\Delta \epsilon_{i0}^{\text{RNAP}} = \Delta \epsilon_{P,i0} \equiv \epsilon_{P,i0}^S - \epsilon_P^{NS}, \quad \Delta \epsilon_{i1}^{\text{RNAP}} = \Delta \epsilon_{P,i1} \equiv \epsilon_{P,i1}^S - \epsilon_P^{NS}, \quad \Delta \epsilon_{i1} \equiv \epsilon_{y1}^S - \epsilon_{y1}^{NS}$$

Suppose that we have p RNA polymerase molecules and r molecules of gene product 1 (the regulator). We model the genome as a “reservoir” with n nonspecific binding sites (to which either RNAP or regulator can bind). One of these sites is the promoter of gene i . Three different classes of configurations interest us:

1. empty promoter
2. regulator bound to promoter
3. regulator and RNAP bound to promoter
4. RNAP only bound to promoter

These correspond to the following partial partition functions, which represent the “unnormalized probabilities” of each configuration.

1. $Z(p, r)$
2. $Z(p, r-1) e^{-\beta \epsilon_{i1}^S}$
3. $Z(p-1, r-1) e^{-\beta \epsilon_{i1}^S} e^{-\beta \epsilon_{P,i1}^S}$
4. $Z(p-1, r) e^{-\beta \epsilon_{P,i0}^S}$

where $Z(p, r) = \frac{n!}{p!r!(n-p-r)!} e^{-\beta r \epsilon_{i1}^{NS}} e^{-\beta \epsilon_P^{NS}}$.

$Z(p, r)$ is equal to the total number of arrangements of RNAP and y_1 on the nonspecific binding sites times the Boltzmann factor, which gives the relative probability $e^{-\beta \epsilon}$ of a particular state in terms of its energy ϵ .

Since RNAP binds the promoter only in the third and fourth classes of configurations, the probability that RNAP binds the promoter is equal to the unnormalized probability of the third and fourth configurations divided by the “total probability” (the sum of the unnormalized probabilities of all classes of configurations). Hence

$$\begin{aligned}
p_{\text{bound}} &= \frac{Z(p-1, r)e^{-\beta\epsilon_{P,i0}^S} + Z(p-1, r-1)e^{-\beta\epsilon_{i1}^S}e^{-\beta\epsilon_{P,i1}^S}}{Z(p, r) + Z(p-1, r)e^{-\beta\epsilon_{P,i0}^S} + Z(p, r-1)e^{-\beta\epsilon_{i1}^S} + Z(p-1, r-1)e^{-\beta\epsilon_{i1}^S}e^{-\beta\epsilon_{P,i1}^S}} \\
&\approx \frac{\frac{n^{p-1}n^r}{(p-1)!r!}e^{-\beta r\epsilon_{i1}^{NS}}e^{-\beta(p-1)\epsilon_P^{NS}}e^{-\beta\epsilon_{P,i0}^S} + \frac{n^{p-1}n^{r-1}}{(p-1)!(r-1)!}e^{-\beta(r-1)\epsilon_{i1}^{NS}}e^{-\beta(p-1)\epsilon_P^{NS}}e^{-\beta\epsilon_{i1}^S}e^{-\beta\epsilon_{P,i1}^S}}{\frac{n^p n^r}{p!r!}e^{-\beta r\epsilon_{i1}^{NS}}e^{-\beta p\epsilon_P^{NS}} + \frac{n^{p-1}n^r}{(p-1)!r!}e^{-\beta r\epsilon_{i1}^{NS}}e^{-\beta(p-1)\epsilon_P^{NS}}e^{-\beta\epsilon_{P,i0}^S} + \dots} \\
&= \frac{\frac{p}{n}e^{\beta\epsilon_P^{NS}}e^{-\beta\epsilon_{P,i0}^S} + \frac{p}{n}\frac{r}{n}e^{\beta\epsilon_{i1}^{NS}}e^{\beta\epsilon_P^{NS}}e^{-\beta\epsilon_{i1}^S}e^{-\beta\epsilon_{P,i1}^S}}{1 + \frac{p}{n}e^{\beta\epsilon_P^{NS}}e^{-\beta\epsilon_{P,i0}^S} + \frac{r}{n}e^{\beta\epsilon_{i1}^{NS}}e^{-\beta\epsilon_{i1}^S} + \frac{p}{n}\frac{r}{n}e^{\beta\epsilon_{i1}^{NS}}e^{\beta\epsilon_P^{NS}}e^{-\beta\epsilon_{i1}^S}e^{-\beta\epsilon_{P,i1}^S}} \\
&= \frac{\frac{p}{n}e^{-\beta\Delta\epsilon_{P,i0}} + \frac{p}{n}\frac{r}{n}e^{-\beta\Delta\epsilon_{i1}}e^{-\beta\Delta\epsilon_{P,i1}}}{1 + \frac{p}{n}e^{-\beta\Delta\epsilon_{P,i0}} + \frac{r}{n}e^{-\beta\Delta\epsilon_{i1}} + \frac{p}{n}\frac{r}{n}e^{-\beta\Delta\epsilon_{i1}}e^{-\beta\Delta\epsilon_{P,i1}}} \\
&= \frac{\frac{p}{n}e^{-\beta\Delta\epsilon_{P,i0}} + \frac{p}{n}\frac{r}{n}e^{-\beta\Delta\epsilon_{i1}}e^{-\beta\Delta\epsilon_{P,i1}}}{1 + \frac{p}{n}e^{-\beta\Delta\epsilon_{P,i0}} + \frac{r}{n}e^{-\beta\Delta\epsilon_{i1}}(1 + \frac{p}{n}e^{-\beta\Delta\epsilon_{P,i1}})} \\
&= \frac{Pe^{-\beta\Delta\epsilon_{i0}^{\text{RNAP}}} + Py_1e^{-\beta\Delta\epsilon_{i1}}e^{-\beta\Delta\epsilon_{i1}^{\text{RNAP}}}}{1 + Pe^{-\beta\Delta\epsilon_{i0}^{\text{RNAP}}} + y_1e^{-\beta\Delta\epsilon_{i1}}(1 + Pe^{-\beta\Delta\epsilon_{i1}^{\text{RNAP}}})},
\end{aligned}$$

where in the second line we used the approximation $\frac{n!}{p!r!(n-p-r)!} \approx \frac{n^p n^r}{p!r!}$ which hold for $p, r \ll n$, in the third we divided by $\frac{n^p n^r}{p!r!}e^{-\beta r\epsilon_{i1}^{NS}}e^{-\beta\epsilon_P^{NS}}$, in the third we used the identities $\Delta\epsilon_{P,i0} = \epsilon_{P,i0}^S - \epsilon_P^{NS}$, $\Delta\epsilon_{P,i1} = \epsilon_{P,i1}^S - \epsilon_P^{NS}$, $\Delta\epsilon_{i1} \equiv \epsilon_{i1}^S - \epsilon_{i1}^{NS}$, and in the last we substituted in the definitions $\frac{p}{n} = P$, $\frac{r}{n} = y_1$, $\Delta\epsilon_{i0}^{\text{RNAP}} = \Delta\epsilon_{P,i0}$, $\Delta\epsilon_{i1}^{\text{RNAP}} = \Delta\epsilon_{P,i1}$.

APPENDIX B: NONIDENTIFIABILITY

To see that the equations

$$\frac{dx_i}{dt} = \frac{b_{i0} + \sum_{j=1}^N b_{ij}\Pi_{k \in S_{ij}}x_k}{1 + \sum_{j=1}^N c_{ij}\Pi_{k \in S_{ij}}x_k} - \gamma_i x_i, \quad b_{i0} < 1$$

and

$$\frac{dx_i}{dt} = \frac{(cb_{i0} + \gamma_i)x_i + \sum_{j=1}^N cb_{ij}\Pi_{k \in S_{ij}}x_i x_k}{1 + cx_i + \sum_{j=1}^N cc_{ij}\Pi_{k \in S_{ij}}x_i x_k} - \gamma_i x_i, \quad c \geq \frac{\gamma}{1 - b_{i0}}$$

reduce to the same equation at any steady-state where $x_i \neq 0$, choose any $0 \leq b_{i0} < 1$, $0 \leq b_{ij} \leq c_{ij}$, $1 < j \leq N$, $c \geq \frac{\gamma}{1-b_{i0}}$, and calculate:

$$\begin{aligned}
0 &= \frac{(cb_{i0} + \gamma_i)x_i + \sum_{j=1}^N cb_{ij}\Pi_{k \in S_{ij}}x_ix_k}{1 + cx_i + \sum_{j=1}^N cc_{ij}\Pi_{k \in S_{ij}}x_ix_k} - \gamma_ix_i \\
\iff 0 &= (cb_{i0} + \gamma_i)x_i + \sum_{j=1}^N cb_{ij}\Pi_{k \in S_{ij}}x_ix_k - \gamma_ix_i(1 + cx_i + \sum_{j=1}^N cc_{ij}\Pi_{k \in S_{ij}}x_ix_k) \\
&= cx_i(b_{i0} + \sum_{j=1}^N b_{ij}\Pi_{k \in S_{ij}}x_k - \gamma_ix_i(1 + \sum_{j=1}^N c_{ij}\Pi_{k \in S_{ij}}x_k)) \\
\iff 0 &= b_{i0} + \sum_{j=1}^N b_{ij}\Pi_{k \in S_{ij}}x_k - \gamma_ix_i(1 + \sum_{j=1}^N c_{ij}\Pi_{k \in S_{ij}}x_k)x_i \quad (\text{provided } x_i \neq 0) \\
\iff 0 &= \frac{b_{i0} + \sum_{j=1}^N b_{ij}\Pi_{k \in S_{ij}}x_k}{1 + \sum_{j=1}^N c_{ij}\Pi_{k \in S_{ij}}x_k} - \gamma_ix_i
\end{aligned}$$

APPENDIX C: TIE-BREAKING

Assuming that we allow only first and second-order terms, we can determine whether a given equation is ambiguous as follows. If it includes no self-regulation at all, it is of the simple form, and has a class of alternatives of the higher-order form parametrized by $c \geq \frac{\gamma}{1-b_{i0}}$. On the other hand, if it includes self-regulation in every term except for the constant 1 in the denominator, and the coefficient of x_i in the denominator is greater than γ_i , then it is of the higher-order form and has an alternative of the simple form.

Practically, the simplest way to make this decision is solve the optimization problem 4 twice: once normally, and once without allowing self-regulation (that is, adding the additional constraints that $b_{ij} = 0$ whenever x_i is in the j th complex). We can compare the forms of the recovered equations as well as the quality of the fit (i.e. the unregularized objective). If the equation is ambiguous, the restriction on self-regulation will have little effect on the quality of fit, since it will simply cause the algorithm to choose the simple alternative. Comparing the recovered equations, we will also notice that they are either the same, or have the relationship given by equations 5 and 6. On the other hand, if the equation is unambiguous, the first recovered equation will not have the form of either equation 5 or equation 6, and the quality of the fit will be significantly worse when self-regulation is restricted. This test may not always be conclusive (for example, this occurs for the Nanog and Gata6 equations in the noisy simulation), but if we are unsure we can always apply the derivative tie-breaker described below to both versions of the ambiguous equation as well the equation recovered normally, and select the form that makes the best prediction.

In order to choose between the possible forms of an ambiguous equation (and possibly find c), we can measure the derivative $\frac{dx_i}{dt}$ experimentally and check whether it agrees with the value predicted by the simple form of the equation. Specifically, we can choose and perform a perturbation that is likely to have a major impact on the system, measure the concentrations shortly afterwards, and approximate the derivative by:

$$\frac{dx_i}{dt}(t_0) \approx \frac{x_i(t_1) - x_i(t_0)}{t_1 - t_0}$$

(This type of experiment is not easy to carry out on a large scale, so we must choose which derivatives to measure with care, and do so only when necessary.) Next we predict the derivative following the perturbation using the simple equation. For example, if we knock out term $\Pi_{k \in S_{iJ}} x_k$ starting from steady-state μ , the simple form predicts:

$$\frac{dx_i}{dt} = \frac{b_{i0} + \sum_{j \neq J} b_{ij} \Pi_{k \in S_{ij}} \mu_k}{1 + \sum_{j \neq J} c_{ij} \Pi_{k \in S_{ij}} \mu_k} - \gamma_i \mu_i$$

If the measured and predicted derivatives agree, we know that the simple form is correct. Otherwise, we conclude that the true equation has the higher-order form, and estimate c as follows:

$$(11) \quad c = \frac{-\frac{dx_i}{dt}}{(\frac{dx_i}{dt} + \gamma_i x_i)(x_i + \sum_{j=1}^N c_{ij} \Pi_{k \in S_{ij}} x_i x_k) - b_{i0} x_i - \sum_{j=1}^N b_{ij} \Pi_{k \in S_{ij}} x_i x_k}$$

From a practical perspective, the measured derivative will not agree exactly with the predicted derivative, so if we are unsure whether we have a match, we can solve for c and determine whether it is possible ($c \geq \frac{\gamma}{1-b_{i0}}$) and reasonable. Furthermore, if we are unsure whether or not the equation is ambiguous, we can also predict the derivative with the equation recovered without restrictions, and compare this prediction with those of the two alternative equations.

APPENDIX D: SIMULATED SIX-GENE SUBNETWORK IN MOUSE ESC

We test our method on a synthetic network governed by the system of ODEs given in (7). The $\frac{d[C]}{dt}$, $\frac{d[Gc]}{dt}$, and $\frac{d[G]}{dt}$ equations are ambiguous with the alternative forms given in (8) (provided we ignore the very small constant term in the $\frac{d[C]}{dt}$ equation and $[G]$ term in the $\frac{d[G]}{dt}$). The $\frac{d[C]}{dt}$ equation has the higher-order form and an alternative simple form, while the $\frac{d[Gc]}{dt}$, and $\frac{d[G]}{dt}$ equations have the simple form and alternative higher-order forms. (Although $[S]$ appears in every term of the $\frac{d[S]}{dt}$ equation, it is not ambiguous since $cb_{i0} + \gamma_i = 0$, which is impossible since $c > 0, b_{i0} \geq 0$.) We apply our method twice, once allowing self-regulation and again disallowing it. Then we compare the two recovered forms of each equation and their fit to the data to determine whether nonidentifiability exists in each case. If so, we break the tie by examining derivatives. We do this for both noiseless and noisy data.

Without noise, we use a total of 52 measurements: the expression levels at each of the system steady-states (ESC, DSC, Endo and Trophect) and expression levels at the steady states reached after overexpressing each gene at twice its steady-state level, and knocking it down to one-fifth of its steady-state level, starting from each basic steady-state. We use cross validation (CV) to select the sparsity parameter λ for each equation, with and without self-regulation (Figure 5). We use CVX software to solve the convex optimization problem (Grant and Boyd, 2011, 2008). When we solve without restricting self-regulation (using the sparsity parameters chosen by CV), we recover the following equations (with coefficients thresholded at 0.1% of the largest recovered coefficient):

TABLE 1
Quality of fit (unregularized objective value) for noiseless data

Equation	unrestricted solution	no self-regulation
Oct4	1.249×10^{-5}	5.7674
Sox2	1.1226×10^{-9}	0.4269
Nanog	6.7426×10^{-8}	0.7664
Cdx2	3.5627×10^{-7}	9.7365×10^{-7}
Gcnf	1.2954×10^{-7}	2.130×10^{-7}
Gata6	7.7505×10^{-7}	7.9072×10^{-5}

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{[A] + 0.001 + (0.005[O][S] + 0.025[O][S][N])}{1 + [A] + (0.001[O] + 0.005[O][S] + 0.025[O])[S][N] + 10[O][C] + 10[Gc]} - 0.1[O] \quad (0.01\% \text{ threshold}) \\
\frac{d[S]}{dt} &= \frac{0.001 + 0.005[O][S] + 0.025[O][S][N]}{1 + 0.005[O][S] + 0.025[O][S][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.1[O][S] + 0.1[O][S][N]}{1 + 0.1[O][S] + 0.1[O][S][N] + 10[O][G]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{0.95}{1 + 2.5[O]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.1[Gc] + 0.01[C][Gc] + 0.01[G][Gc]}{1 + 0.1[Gc] + 0.01[C][Gc] + 0.01[G][Gc]} - 0.1[Gc] \\
\frac{d[G]}{dt} &= \frac{0.1 + 0.95[O]}{1 + 0.95[O] + 14.25[N] + 0.04[S][N] + 0.08[N][C] + 0.02[N][Gc] + 0.08[N][G]} - 0.1[G]
\end{aligned}$$

When we solve the same problem, disallowing self-regulation (again using the appropriate CV sparsity parameters), we recover the following equations.

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{[A] + 0.14[C] + 0.57[G] + 0.12[S][G] + 0.04[N][C] + 0.20[N][Gc]}{1 + [A] + 20[C] + 9.6[Gc] + 3.3[G] + 0.33[S][C] + 0.04[N][C] + 0.20[N][Gc]} - 0.1[O] \\
\frac{d[S]}{dt} &= \frac{0.18[O][N]}{1 + 0.18[O][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.01 + 0.12[O][S]}{1 + 0.12[O][S]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{0.95}{1 + 2.5[O]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.001 + 0.1[C] + 0.1[G]}{1 + 0.1[C] + 0.1[G]} - 0.1[Gc] \\
\frac{d[G]}{dt} &= \frac{0.1 + [O]}{1 + [O] + 0.03[N][Gc] + 15[N]} - 0.1[G]
\end{aligned}$$

We measure the quality of the fit by the unregularized objective value $\|G_i b_i - \gamma D_i G_i c_i\|$ from equation 4 for each recovered equation. These values are given in Table 1. We can tell that first three equations are unambiguous, while the last three have alternative forms, since the quality of fit is roughly the same for the last three equations whether or not we restrict self-regulation, while for the first three, the fit is much worse when self-regulation is prohibited. Therefore, we can use

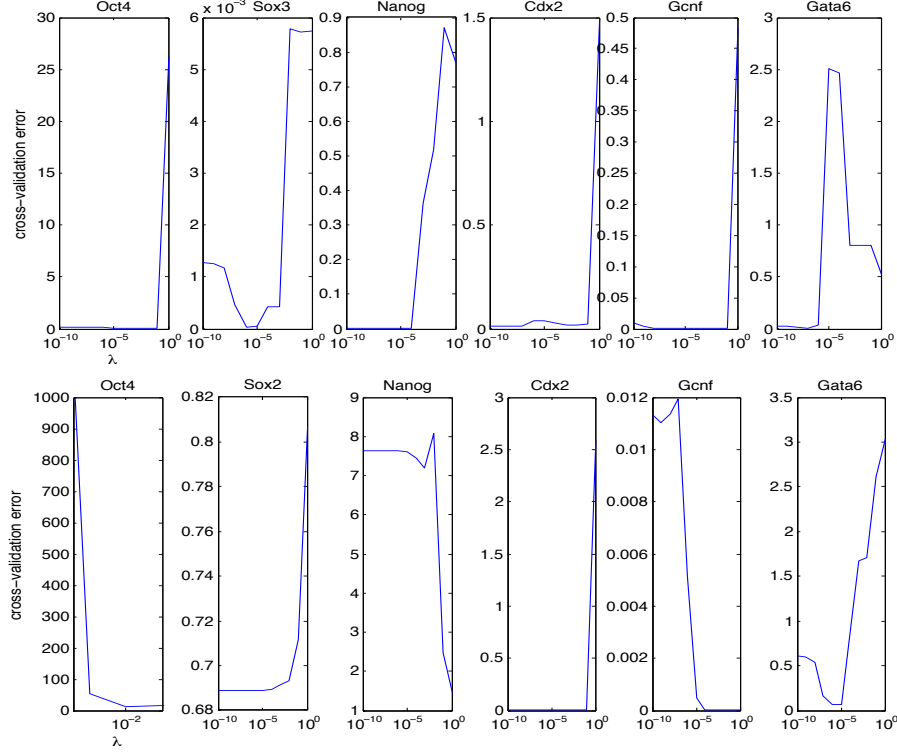


Figure 5: Cross validation (8-fold) on noiseless data. We estimate the test error for each gene equation and various choices of sparsity parameter by randomly dividing the 52 observations into 8 folds (groups), then leaving out each fold out in turn, training the model on the remaining 7 folds, testing on the omitted fold, and finally averaging the 8 resulting test errors. After repeating this process for each gene equation and several choices of sparsity parameter, we select the sparsity parameters corresponding to the lowest error for each equation. (a) Unrestricted case: we selected sparsity parameters $[10^{-1}, 10^{-5}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{-6}]$. (b) No self-regulation: we selected sparsity parameters $[10^{-2}, 10^{-5}, 10^{-1}, 10^{-1}, 10^{-4}, 10^{-5}]$.

the recovered forms of the first three equations, but we need to break ties between alternative forms of the last three equations.

It's easiest to start with the simple forms of the last three equations, and determine the corresponding higher-order forms. To break the tie, we look at derivatives following Oct4, Cdx2, and Nanog knockouts, respectively, since these regulators are important in each of the three ambiguous equations (Figure 6). After each knockout we estimate the derivative with a finite difference, compare it to the derivative prediction made by the simple form of the equation, and accept this form if the match is good. Otherwise we compute c using equation 11, and (provided it is reasonable), accept the higher-order form with this choice of c .

In this case, we estimate $\frac{d[C]}{dt} \approx 0.6004825$ immediately after Oct4 knockout from ESC steady-state. The simple form of the equation yields $\frac{d[C]}{dt} \approx 0.78$ immediately following the knockout, which is a poor match. Therefore we select the higher-order form and use the measured derivative to compute $c = 2$, which yields $\frac{d[C]}{dt} = 0.60$. For the $\frac{d[Gc]}{dt}$ equation we measure $\frac{d[Gc]}{dt} \approx -0.13$ immediately after Cdx2 knockout from SC, and the simple form is a good match at -0.13 (com-

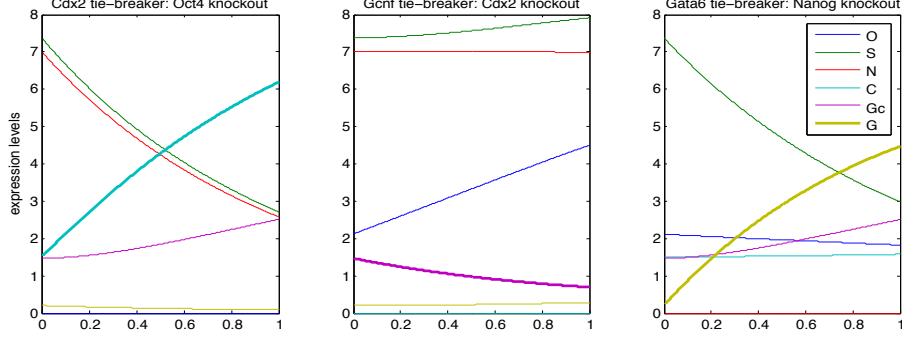


Figure 6: Derivative measurements to break ties between alternative forms. (left) Trajectory of Cdx2 following Oct4 knockout from ESC: $\frac{d[C]}{dt}(t_0) \approx 0.6004825$. (center) Gcnf trajectory following Cdx2 knockout from ESC: $\frac{d[Gc]}{dt}(t_0) \approx -0.1261507$. (right) Gata6 trajectory following Nanog knockout from ESC: $\frac{d[G]}{dt}(t_0) \approx 0.6908821$.

putting c for the higher-order form using the derivative yields an unreasonably large $c \approx 86$; using the minimum value $c = 0.1$ in the higher-order form yields $\frac{d[Gc]}{dt} = -0.017$). Similarly, for the $\frac{d[G]}{dt}$ equation we measure $\frac{d[Gc]}{dt} \approx 0.69$ immediately after Nanog knockout from SC, and the simple form is a good match at 0.69. In the end, we select the equations given in (9).

Next we test the algorithm using noisy data by adding zero-mean Gaussian noise to each measurement, with standard deviation 1% of the measurement magnitude. We use the 4 basic steady-states, the steady states reached after knockdown and overexpression of each individual gene from each basic steady state, and those reached after knocking one gene up and one gene down from each pair of genes, starting from ESC and DSC. Again, we use cross validation to select the sparsity parameters (Figure 7). When we solve the problem with noisy data (with no restriction on self-regulation) and threshold at a level of 1% of the largest recovered coefficient, we recover:

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{[A]}{1 + [A] + 9.9[Gc] + 9.9[O][C]} - 0.1[O] \\
\frac{d[S]}{dt} &= \frac{0.001[O][S] + 0.0005[S][N] + 0.025[O][S][N]}{1 + 0.001[O][S] + 0.0005[S][N] + 0.025[O][S][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.09[O][S][N]}{1 + 0.1[G][Gc] + 0.09[O][S][N] + 9.1[O][G]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{0.94}{1 + 2.4[O]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.1[Gc] + 0.01[C][Gc] + 0.01[G][Gc]}{1 + 0.1[Gc] + 0.01[C][Gc] + 0.01[G][Gc]} - 0.1[Gc] \\
\frac{d[G]}{dt} &= \frac{0.1[G] + 0.1[O][G]}{1 + 0.2[N] + 0.1[G] + 0.05[O][N] + 0.1[O][G] + 0.05[S][N] + 0.025[N][C] + 1.4[N][G]} - 0.1[G]
\end{aligned}$$

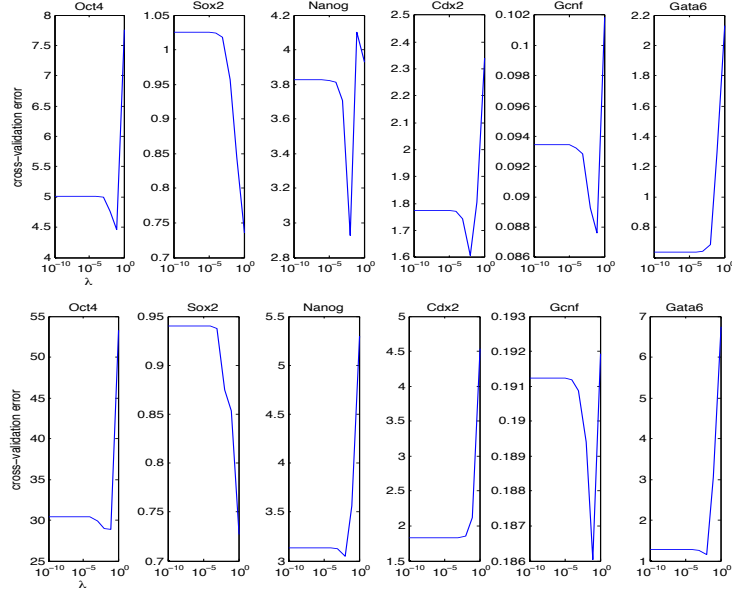


Figure 7: Cross validation (8-fold on 108 observations, with the approach described in Figure 5) on noisy data (1% Gaussian noise): (left) unrestricted: we selected sparsity parameters $[0.1, 1, 0.01, 0.01, 0.1, 0.00001]$ (right) No self-regulation: we selected $[0.1, 1, 0.01, 0.001, 0.1, 0.01]$.

When we solve without allowing self-regulation, we recover:

$$\begin{aligned}
\frac{d[O]}{dt} &= \frac{[A] + 0.3[G]}{1 + [A] + 15.4[C] + 9.7[Gc] + 3.1[G] + 0.5[S][C] + 0.6[N][C]} - 0.1[O] \\
\frac{d[S]}{dt} &= \frac{0.2[O][N]}{1 + 0.2[O][N]} - 0.1[S] \\
\frac{d[N]}{dt} &= \frac{0.03 + 0.17[S] + 0.03[S][C]}{1 + 0.17[S] + 0.03[S][C]} - 0.1[N] \\
\frac{d[C]}{dt} &= \frac{0.95}{1 + 2.5[O]} - 0.1[C] \\
\frac{d[Gc]}{dt} &= \frac{0.1[C] + 0.1[G]}{1 + 0.1[C] + 0.1[G]} - 0.1[Gc] \\
\frac{d[G]}{dt} &= \frac{0.1 + 0.9[O]}{1 + 0.9[O] + 14.2[N]} - 0.1[G]
\end{aligned}$$

Table 2 shows that for noisy data, the quality of fit does not indicate as clearly which equations are ambiguous. For the Oct4 and Sox2 equations, the quality of fit drops dramatically when we restrict self-regulation, while it changes very little for Cdx2, Gcnf and Gata6, revealing that the Oct4 and Sox2 equations are correct, while Cdx2, Gcnf and Gata6 have a simple and a higher-order form. We break the tie between the two forms of the last three equations using derivatives as before. The Nanog equation is unclear, so we analyze derivatives to decide between the two alternative forms *and* the solution of the unrestricted optimization problem. First we observe that the higher-order version of the ambiguous form is illegal as it contains third-order terms, so we only need to choose between the unrestricted equation and the simple equation recovered without self-regulation. We

TABLE 2
Quality of fit (unregularized objective value) for noisy data

Equation	unrestricted	no self-reg
Oct4	1.5170	8.1241
Sox2	0.2800	1.006
Nanog	0.6877	1.9347
Cdx2	0.5634	0.8208
Gcnf	0.0278	0.0599
Gata6	0.1278	0.2224

simulate the trajectory after Gata6 knockout from ESC and compare the derivative ($\frac{d[N]}{dt} = 0.023$) to the predictions of the first equation ($\frac{d[N]}{dt} = 0.044$) and the simple version ($\frac{d[N]}{dt} = -0.070$), concluding that the first equation is correct. Finally, we obtain the equations given in equation 10.

ACKNOWLEDGEMENTS

Many thanks to Xi Chen for helpful discussions. This work was partially supported by NIH grant R01-HG00601801 and NSF grant DMS-0906044 to WHW and NIH grant GM007276 to YL.

REFERENCES

- ACKERS, G. K., JOHNSON, A. D. and SHEA, M. A. (1982). Quantitative model for gene regulation by lambda phage repressor. *Proc. Natl. Acad. Sci. U.S.A.* **79** 1129–1133.
- ALON, U. (2007). Network motifs: theory and experimental approaches. *Nat. Rev. Genet.* **8** 450–461.
- AVERY, L. and WASSERMAN, S. (1992). Ordering gene function: the interpretation of epistasis in regulatory hierarchies. *Trends Genet.* **8** 312–316.
- BANSAL, M., BELCASTRO, V., AMBESI-IMPIOMBATO, A. and DI BERNARDO, D. (2007). How to infer gene networks from expression profiles. *Mol. Syst. Biol.* **3** 78.
- BAR-JOSEPH, Z., GERBER, G. K., LEE, T. I., RINALDI, N. J., YOO, J. Y., ROBERT, F., GORDON, D. B., FRAENKEL, E., JAAKKOLA, T. S., YOUNG, R. A. and GIFFORD, D. K. (2003). Computational discovery of gene modules and regulatory networks. *Nat. Biotechnol.* **21** 1337–1342.
- BINTU, L., BUCHLER, N. E., GARCIA, H. G., GERLAND, U., HWA, T., KONDEV, J., KUHLMAN, T. and PHILLIPS, R. (2005a). Transcriptional regulation by the numbers: applications. *Curr. Opin. Genet. Dev.* **15** 125–135.
- BINTU, L., BUCHLER, N. E., GARCIA, H. G., GERLAND, U., HWA, T., KONDEV, J. and PHILLIPS, R. (2005b). Transcriptional regulation by the numbers: models. *Curr. Opin. Genet. Dev.* **15** 116–124.
- CHICKARMANE, V. and PETERSON, C. (2008). A computational model for understanding stem cell, trophoderm and endoderm lineage determination. *PLoS ONE* **3** e3478.
- CHOI, B. (2012). Learning Networks in Biological Systems, Ph.D. thesis, Department of Applied Physics, Stanford University, Stanford, California (thesis supervisor: W.H.Wong).
- CRICK, F. (1970). Central dogma of molecular biology. *Nature* **227** 561–563.
- DE SMET, R. and MARCHAL, K. (2010). Advantages and limitations of current network inference methods. *Nat. Rev. Microbiol.* **8** 717–729.
- DERISI, J. L., IYER, V. R. and BROWN, P. O. (1997). Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science* **278** 680–686.
- EISEN, M. B., SPELLMAN, P. T., BROWN, P. O. and BOTSTEIN, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. U.S.A.* **95** 14863–14868.
- FAITH, J. J., HAYETE, B., THADEN, J. T., MOGNO, I., WIERZBOWSKI, J., COTTAREL, G., KASIF, S., COLLINS, J. J. and GARDNER, T. S. (2007). Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol.* **5** e8.
- FRIEDMAN, N. (2004). Inferring cellular networks using probabilistic graphical models. *Science* **303** 799–805.
- GARCIA, H. G., KONDEV, J., ORME, N., THERIOT, J. A. and PHILLIPS, R. (2011). Thermodynamics of biological processes. *Meth. Enzymol.* **492** 27–59.
- GARDNER, T. S., DI BERNARDO, D., LORENZ, D. and COLLINS, J. J. (2003). Inferring genetic networks and identifying compound mode of action via expression profiling. *Science* **301** 102–105.

- GRANT, M. and BOYD, S. (2008). Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*, (V. Blondel, S. Boyd and H. Kimura, eds.). *Lecture Notes in Control and Information Sciences* 95–110. Springer-Verlag Limited http://stanford.edu/~boyd/graph_dcp.html.
- GRANT, M. and BOYD, S. (2011). CVX: Matlab Software for Disciplined Convex Programming, version 1.21.
- HOLSTEGE, F. C., JENNINGS, E. G., WYRICK, J. J., LEE, T. I., HENGARTNER, C. J., GREEN, M. R., GOLUB, T. R., LANDER, E. S. and YOUNG, R. A. (1998). Dissecting the regulatory circuitry of a eukaryotic genome. *Cell* **95** 717–728.
- HU, Z., KILLION, P. J. and IYER, V. R. (2007). Genetic reconstruction of a functional transcriptional regulatory network. *Nat. Genet.* **39** 683–687.
- HUGHES, T. R., MARTON, M. J., JONES, A. R., ROBERTS, C. J., STOUGHTON, R., ARMOUR, C. D., BENNETT, H. A., COFFEY, E., DAI, H., HE, Y. D., KIDD, M. J., KING, A. M., MEYER, M. R., SLADE, D., LUM, P. Y., STEPANIAN, S. B., SHOEMAKER, D. D., GACHOTTE, D., CHAKRABURTTY, K., SIMON, J., BARD, M. and FRIEND, S. H. (2000). Functional discovery via a compendium of expression profiles. *Cell* **102** 109–126.
- JACOB, F. and MONOD, J. (1961). Genetic regulatory mechanisms in the synthesis of proteins. *J. Mol. Biol.* **3** 318–356.
- LEE, T. I., RINALDI, N. J., ROBERT, F., ODOM, D. T., BAR-JOSEPH, Z., GERBER, G. K., HANNETT, N. M., HARBISON, C. T., THOMPSON, C. M., SIMON, I., ZEITLINGER, J., JENNINGS, E. G., MURRAY, H. L., GORDON, D. B., REN, B., WYRICK, J. J., TAGNE, J. B., VOLKERT, T. L., FRAENKEL, E., GIFFORD, D. K. and YOUNG, R. A. (2002). Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science* **298** 799–804.
- MARBACH, D., PRILL, R. J., SCHAFFTER, T., MATTIUSI, C., FLOREANO, D. and STOLOVITZKY, G. (2010). Revealing strengths and weaknesses of methods for gene network inference. *Proc. Natl. Acad. Sci. U.S.A.* **107** 6286–6291.
- MORTAZAVI, A., WILLIAMS, B. A., MCCUE, K., SCHAEFFER, L. and WOLD, B. (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods* **5** 621–628.
- PINNA, A., SORANZO, N. and DE LA FUENTE, A. (2010). From knockouts to networks: establishing direct cause-effect relationships through graph analysis. *PLoS ONE* **5** e12912.
- REN, B., ROBERT, F., WYRICK, J. J., APARICIO, O., JENNINGS, E. G., SIMON, I., ZEITLINGER, J., SCHREIBER, J., HANNETT, N., KANIN, E., VOLKERT, T. L., WILSON, C. J., BELL, S. P. and YOUNG, R. A. (2000). Genome-wide location and function of DNA binding proteins. *Science* **290** 2306–2309.
- ROBERTSON, G., HIRST, M., BAINBRIDGE, M., BILENKY, M., ZHAO, Y., ZENG, T., EUSKIRCHEN, G., BERNIER, B., VARHOL, R., DELANEY, A., THIESSEN, N., GRIFFITH, O. L., HE, A., MARRA, M., SNYDER, M. and JONES, S. (2007). Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat. Methods* **4** 651–657.
- ROSENFELD, S. (2011). Mathematical descriptions of biochemical networks: stability, stochasticity, evolution. *Prog. Biophys. Mol. Biol.* **106** 400–409.
- SCHAFFTER, T., MARBACH, D. and FLOREANO, D. (2011). GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics* **27** 2263–2270.
- SEGAL, E., SHAPIRA, M., REGEV, A., PE’ER, D., BOTSTEIN, D., KOLLER, D. and FRIEDMAN, N. (2003). Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat. Genet.* **34** 166–176.
- SHEA, M. A. and ACKERS, G. K. (1985). The OR control system of bacteriophage lambda. A physical-chemical model for gene regulation. *J. Mol. Biol.* **181** 211–230.
- TEGNER, J., YEUNG, M. K., HASTY, J. and COLLINS, J. J. (2003). Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling. *Proc. Natl. Acad. Sci. U.S.A.* **100** 5944–5949.
- TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc. B.* **58** 267–288.
- TYSON, J. J., CHEN, K. C. and NOVAK, B. (2003). Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Curr. Opin. Cell Biol.* **15** 221–231.
- VON HIPPEL, P. H., REVZIN, A., GROSS, C. A. and WANG, A. C. (1974). Non-specific DNA binding of genome regulating proteins as a biological control mechanism: I. The lac operon: equilibrium aspects. *Proc. Natl. Acad. Sci. U.S.A.* **71** 4808–4812.
- YIP, K. Y., ALEXANDER, R. P., YAN, K. K. and GERSTEIN, M. (2010). Improved reconstruction of in silico gene regulatory networks by integrating knockout and perturbation data. *PLoS ONE* **5** e8121.
- ZHOU, Q., CHIPPERFIELD, H., MELTON, D. A. and WONG, W. H. (2007). A gene regulatory network in mouse embryonic stem cells. *Proc. Natl. Acad. Sci. U.S.A.* **408** 16438–16443.

318 CAMPUS DRIVE
 STANFORD UNIVERSITY, CA 94305
 E-MAIL: whwong@stanford.edu